

ODBMS.ORG User Report No.2/08

Editor Roberto V. Zicari- www.odbms.org

July 2008.

Category: Academia

Domain: Research/Education

User Name: Pieter van Zyl

Title: Researcher

Organization: [Meraka Institute](#) of South Africa's Council for Scientific and Industrial Research (CSIR) and University of Pretoria, South Africa.

Pieter van Zyl is a full time software developer and researcher at the [Meraka Institute](#) of South Africa's Council for Scientific and Industrial Research (CSIR). He is also busy obtaining his Masters in Computer Science at the University of Pretoria. He is involved with the OSS projects [Dithaka](#), [PolePosition](#), [Coefficient](#), and [DictionaryMaker](#). His research focusses on object persistence mechanisms (ORM tools and object databases), with a specific focus on performance benchmarks. He is part of the [Espresso research group](#) at the University of Pretoria.

Q1. Please explain briefly what are your application domains and your role in the enterprise.

Pieter van Zyl : I have developed software for the following domains: military logistics, financial/banking, medical.

I am currently working for a research institute where we develop applications for researchers with a focus on developing Open Source software and to use Open Source as a means to share knowledge with communities.

I have developed web based applications using Java EE, JSF, SEAM, Velocity, Tapestry, Struts Hibernate and JPA. My current work assignments are mobile development (Java ME) and standalone GUI (Swing) applications.

Q2. When the data models used to persistently store data (whether file systems or database management systems) and the data models used to write programs against the data (C++, Smalltalk, Visual Basic, Java, C#) are different, this is referred to as the "impedance mismatch" problem. Do you have an "impedance mismatch" problem?

Pieter van Zyl : Yes, I have experienced this mismatch during development. It is always frustrating to have nice Java POJO's representing my object model and then trying to store and retrieve them

from the relational database. Mapping associations from the object model to relational is always time consuming and requires much thought.

Q3. What solution(s) do you use for storing and managing persistence objects? What experience do you have in using the various options available for persistence for new projects? What are the lessons learned in using such solution(s)?

Pieter van Zyl : I have used ORM mapping tools on top of relational databases or straight object databases to persist objects.

I have used JPA and Hibernate for the past two years and found a vast improvement in persistence and development time using JPA in the Java EE space. JPA coupled with SEAM makes persistence in the web environment so much easier.

A pivotal part of my Masters Degree research is on performance of object persistence between commercial and open source databases. So far I would say that commercial object databases is performing better than open source object databases, in the enterprise space where there are large datasets involved. As for open source databases, I like db4o for its simplicity. Currently db4o seems to be more geared towards the embedded environment and it might grow into an enterprise database.

Some lessons and issues to keep in mind are:

- Keeping track of attached/detached objects are tricky. It is also necessary to know if objects are accessed inside a transaction or not. Some persistence mechanisms make it easier for the developer and others will not.
- Lazy loading issues and the need for good persistent collections that provide lazy loading.
- Mapping one-to-many and many-to-many is time consuming. Other questions that are important: do you need to include a reference back to the parent or not? Which mapping strategy would be best for your model? Which annotation to use?
- Identity management. Do you use the business logic identity? Would you use Java hash code or the equals() methods for identity? Or some other form of identity?
- Using the right tool for the job is important.
- Object Databases differ from each other in functionality and performance
- Some databases perform well with queries and others with traversals.

- It is important to test your application and know if it is query or traversal intensive.

Q4. Do you believe that Object Database systems are a suitable solution to the "object persistence" problem? If yes why? If not, why?

Pieter van Zyl : Yes I believe they are. ODBMS functionality are just as good as relational databases and their performance is very good. In the end object databases and relational databases are database management systems. Both relation and object databases provide scheme management, index management, concurrency control, cache management, etc

I don't agree in creating a new ODMG standard based on SBA. The basis for object databases is object oriented theory and that is a good enough. If you write a ODBMS for Java you should use its model and its language specifications and then inject the code necessary for persistence and retrieval.

Q5. What would you wish as new research/development in the area of Object Persistence in the next 12-24 months?

Pieter van Zyl : New development:

- Improved open source tools for open source databases
- I would like to see a database like Gemstone Facets open sourced and updated
- A new version of JDO or JPA that include or matches today's object databases
- An easier mechanism for relational and ORM developers to switch to an ODBMS.
- Teaching students more about object databases and persistence techniques

New research:

- I have read a few papers where they have used an ODBMS with ontologies. A research question based on this would be: Is odbms only just a persistence tool or is it the best match for ontologies?
- A large research project comparing all the ODBMS's and relational databases performance by:
 - Documenting architectures used by the databases
 - Large scale database performance
 - and including me to participate in this study